



المدرسة الوطنية العليا في الأمن السيبراني  
NATIONAL SCHOOL OF CYBERSECURITY

## 1ST YEAR BASIC TRAINING IN CYBER SECURITY

# INTRODUCTION TO OPERATING SYSTEMS 1 (SYST1)

Dr. Sassi BENTRAD

✉ : [sassi.bentrad.enscs@gmail.com](mailto:sassi.bentrad.enscs@gmail.com) || [sassi.bentrad@enscs.edu.dz](mailto:sassi.bentrad@enscs.edu.dz)

LISCO Laboratory (Laboratoire d'Ingénierie des Systèmes COMplexes) / University of Badji Mokhtar-Annaba (UBMA)  
National School of Cyber Security (NSCS)

Basic Training in Cyber Security (1BT)  
Formation de Base en Cyber-Sécurité (1FB)



## CHAPTER

1

### INTRODUCTION TO OPERATING SYSTEMS

SYST1'2025/2026



## COURSE CONTENT

### CHAPTER 1

#### INTRODUCTION TO OPERATING SYSTEMS (15 %)

- ❑ General introduction
- ❑ Definitions : Computer, Computer System, Operating System,  
...
- ❑ Operating system architecture
- ❑ Operating system functions
- ❑ Types of operating systems
- ❑ Historical overview of operating systems
- ❑ Terminology : Kernel, System Call, Processes, Resources, ...
- ❑ Summary & Key Insights

 GENERAL INTRODUCTION ... Introduction

"The **Operating Systems (OS)** is the **BACKBONE** of all computing systems — enabling communication between user-level applications and hardware."

- **Operating Systems (OS)** are the **foundation of all computing environments**, providing the essential interface between **hardware and users/applications**.
- **Without an OS**, a computer system cannot function efficiently, manage resources, or offer user interaction.
  - 📌 The **OS** is essential for:
    - Managing computer **resources**
    - Providing a **user interface**
    - Ensuring **application execution**
    - Enforcing **security & stability**
- This chapter introduces the **core principles, roles, and evolution** of operating systems in both personal and enterprise environments.

 GENERAL INTRODUCTION ... Introduction **Learning Outcomes — By the end of this chapter, you will be able to:**

- **Define** what an **Operating System (OS)** is and explain its fundamental purpose.
- **Describe** the **core functions** of an OS, including process, memory, file, and device management.
- **Understand** the **need for an operating system** as the interface between users, applications, and hardware.
- **Explain** the **basic principles and architecture** of modern operating systems.
- **Identify and differentiate** between the **main types of operating systems** (e.g., batch, multitasking, real-time, distributed).
- **Recognize and discuss** the **historical evolution** of operating systems and their impact on today's computing environments.

## ❖ GENERAL INTRODUCTION ...

### □ Computer System

A **computer system** is a set of equipment and software designed for the **automatic processing of information**.

It performs the following key operations :

- **Recording (Input):** Capturing and storing information.
- **Restitution (Output):** Displaying or returning processed information.
- **Modification (Processing):** Transforming or updating data.
- **Deletion:** Removing or erasing information when necessary.

A complete **computer system** consists of three main components:

- **Hardware:** The physical components of the computer.
- **System Software:** Programs that manage hardware resources (e.g., the OS).
- **Application Software:** Programs that perform specific user tasks.



❖ GENERAL INTRODUCTION ...

□ What is a Computer – **Calculator** ?

A **computer** is a machine that can process information (perform operations, calculations); it is a **calculator** (**HASSOUBE**).

A **simple image of a Computer** is the **calculator**: a machine that allows us to perform calculations - **operations**.

To process information, the computer runs **programs**.

***System programs & Application programs***



**Computer = Calculator**

❖ GENERAL INTRODUCTION ...

□ What is a Computer – **Calculator** ?

A **computer** is a complex configuration of electronic components interconnected on a **motherboard**.

A **computer** is a machine designed to **process information** — such as text, sound, images, and more.

It performs four essential functions :

- **Acquire Information** — input data from various sources
- **Store Information** — keep data for immediate or future use
- **Process Information** — transform data into meaningful results
- **Output / Restore Information** — display or return processed results

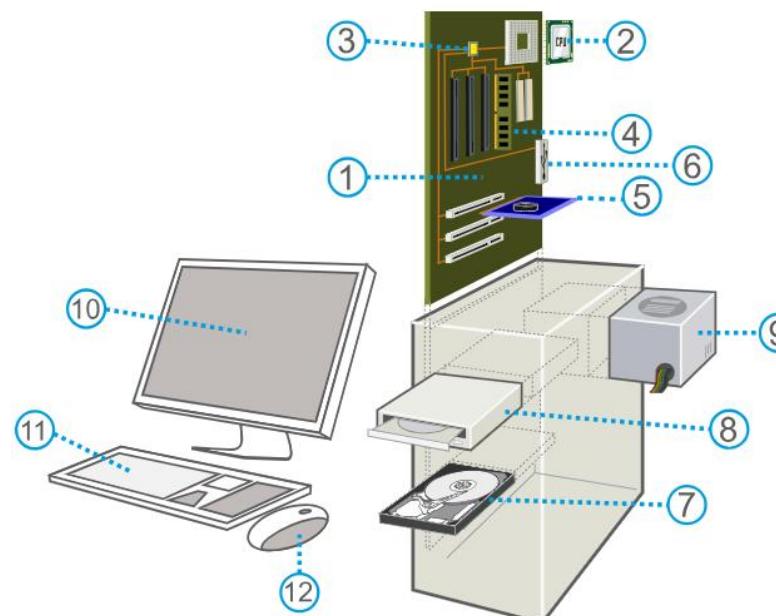


## ❖ GENERAL INTRODUCTION ...

### □ Computer components – *Peripheral Devices*

A **computer** is made up of a **set of interconnected electronic components**.

These elements are placed on a **printed circuit board** known as the **motherboard** (a plate on which the electrical connections linking the components are engraved)....



Main Components of a Personal Computer	
1	The <b>motherboard</b>
2	The <b>processor or microprocessor</b>
3	The <b>bus(es)</b>
4	The <b>RAM (Random Access Memory)</b>
5	The <b>graphics card</b>
6	The <b>input-output devices</b>
7	The <b>hard drive</b>
8	The <b>disk drive</b>
9	The <b>power supply</b>
10	The <b>monitor</b>
11	The <b>keyboard</b>
12	The <b>mouse</b>

**Computer components**  
**(PC : Personal Computer)**

## ❖ GENERAL INTRODUCTION ...

### □ Computer components – *Peripheral Devices*

A computer system is composed of several **fundamental components**, each serving a specific role in processing and managing information:

#### ▪ Central Processing Unit (CPU)

The “brain” of the computer. Executes instructions and controls the operation of all other units.

#### ▪ Main Memory (Primary Memory)

Temporarily stores data and instructions currently in use.  
Provides fast access for the CPU (e.g., RAM).

#### ▪ Storage Units (Secondary Memory)

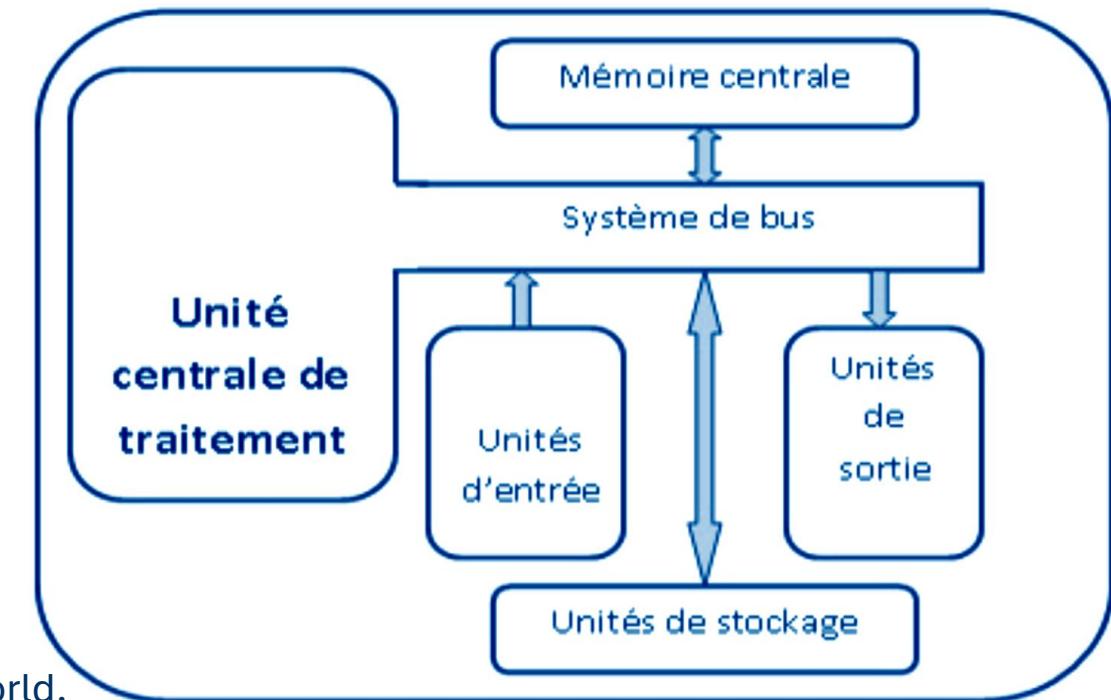
Retains data and programs permanently.  
Examples: Hard drives, SSDs, flash storage.

#### ▪ Input/Output (I/O) Units

Enable communication between the computer and the external world.  
Input: Keyboard, mouse, sensors. Output: Display, printer, actuators.

#### ▪ Bus System

A communication pathway that connects all major components.  
Transfers data, addresses, and control signals between the CPU, memory, and I/O devices.



Basic Structure of a Computer System

## ❖ GENERAL INTRODUCTION ...

## □ Computer components – Peripheral Devices

## ? Problem: Managing Computer Hardware

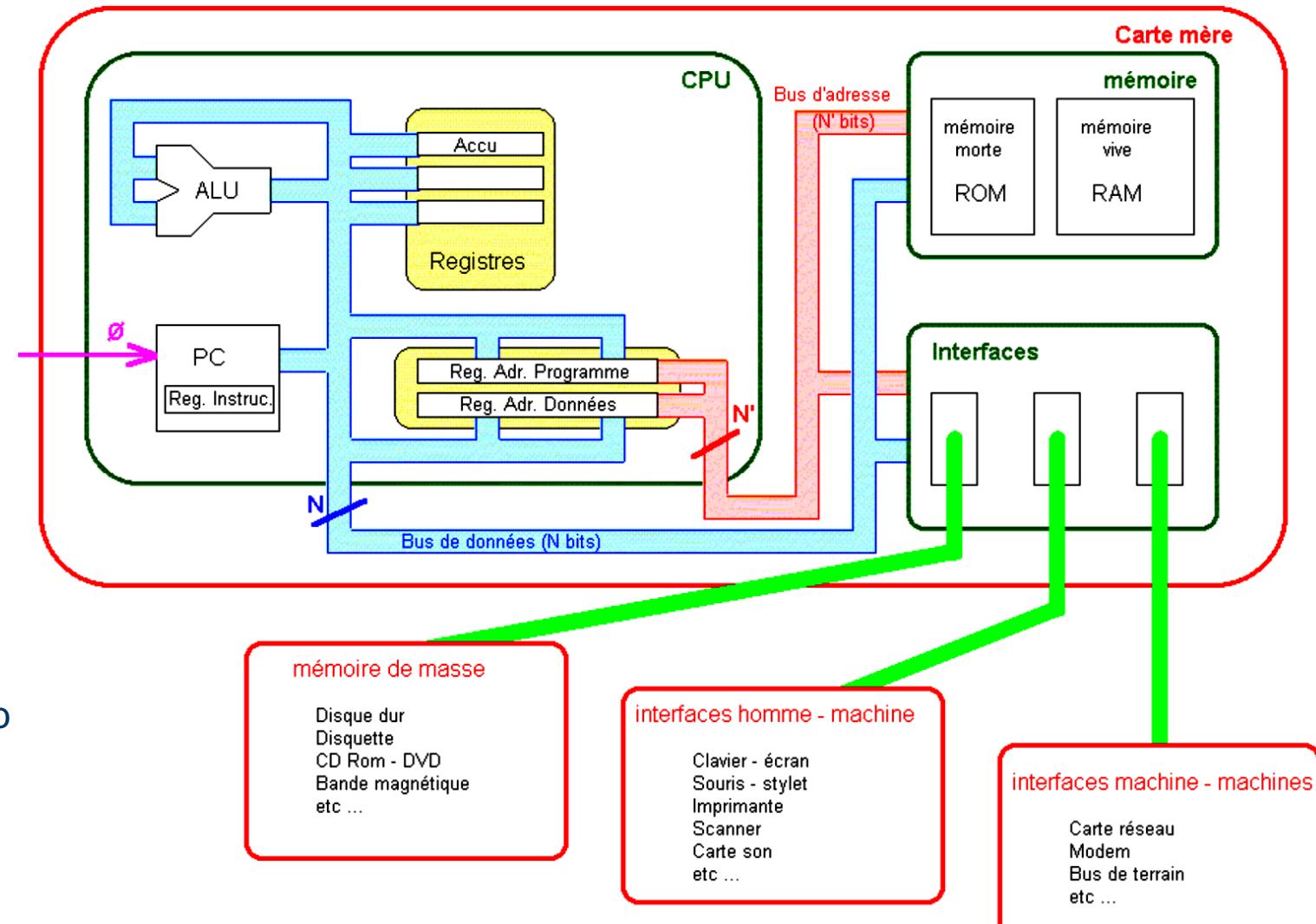
Modern computer systems consist of many physical components — but...

**How can we make all these devices work together?**

- ✓ CPUs, memory, disks, and I/O devices must operate in coordination.
- ✓ Each has its own speed, interface, and control mechanism.

**How can we provide easy and efficient access to these physical resources?**

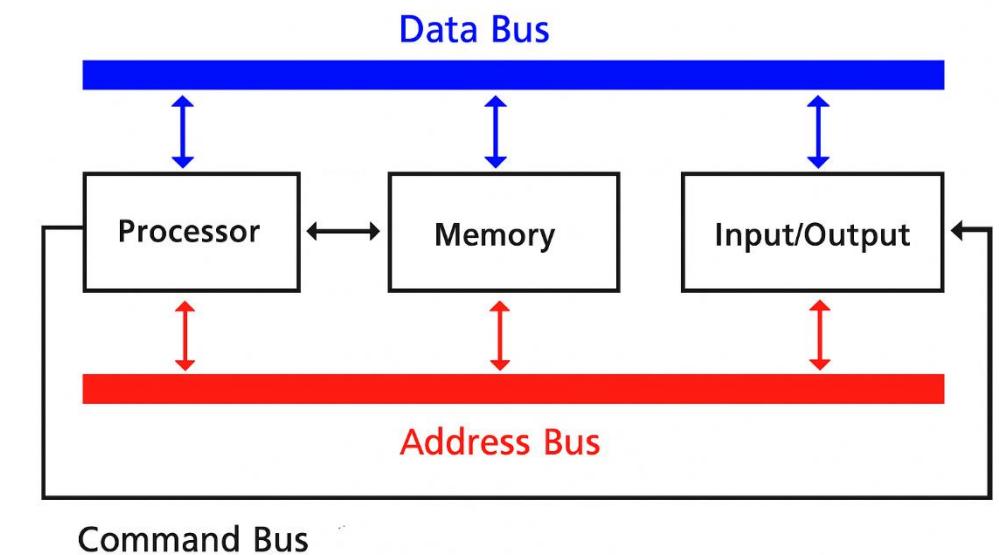
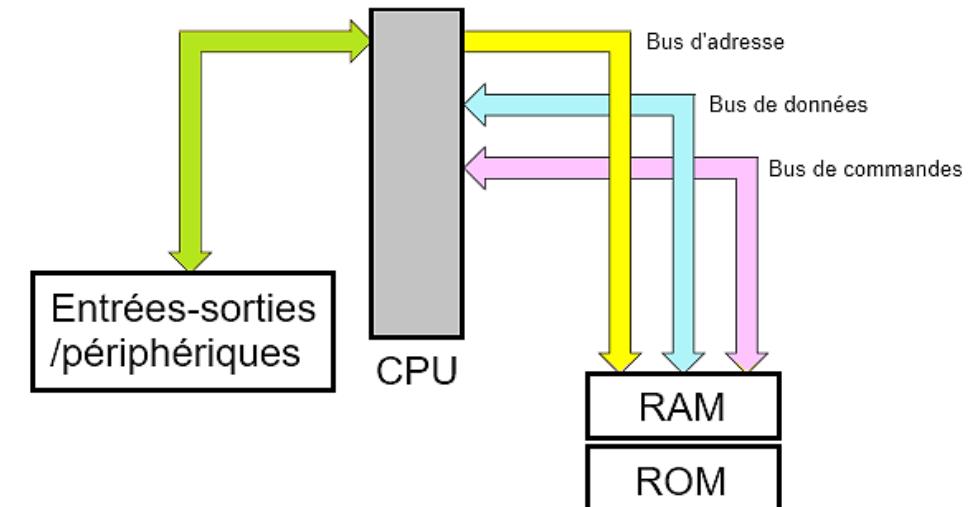
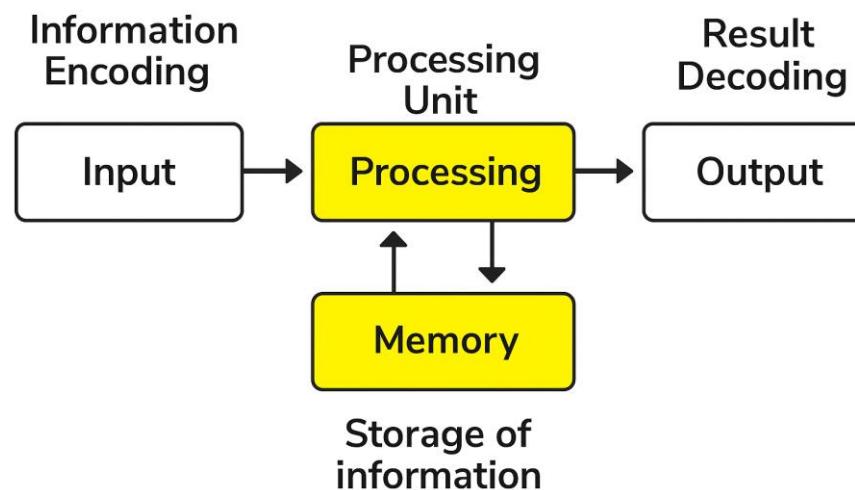
- ✓ Users and applications should not need to manage hardware details.
- ✓ We need a layer that abstracts, manages, and shares these resources safely.



❖ GENERAL INTRODUCTION ...

□ Computer components – *Peripheral Devices*

 This challenge leads to the concept of the *Operating System (OS)*—  
the software responsible for controlling hardware  
and providing a convenient interface  
for users and programs.



 OPERATING SYSTEMS The Foundation of Modern Computing: *Operating Systems*

- A computer and its peripherals form a **computer system** that is very complicated to manage.
- The user always hopes to use the components of a computer correctly and efficiently.

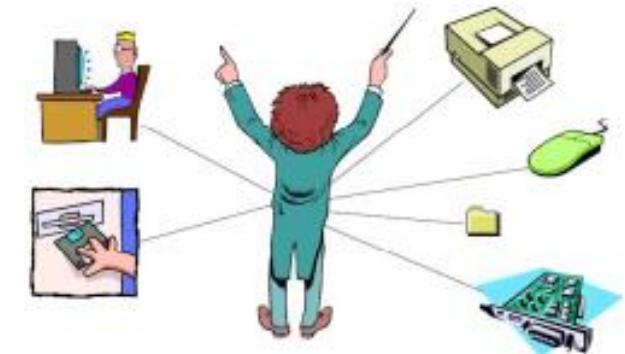
**This is an extremely difficult task for a user.**

- ✓ For this reason, computers must be equipped with an **operating system**.
- ✓ The **operating system** is the first program to be run when the machine is started;
- ✓ Once the operating system has been started, the computer and its peripheral devices will be ready for use by the user (typing on the keyboard, displaying on the screen, printing on the printer, using the mouse, etc.).

## ❖ OPERATING SYSTEMS

### □ The Foundation of Modern Computing: *Operating Systems*

- **System** = set of elements (**programs** and **processes**) that interact with each other according to certain rules.
- **Operating** = Action of exploiting → using to make a profit.
- These elements make the computer **easy to use**, and ensure that hardware resources (*CPU, Memory, etc.*) are used **efficiently** and **economically**.



## ❖ OPERATING SYSTEMS

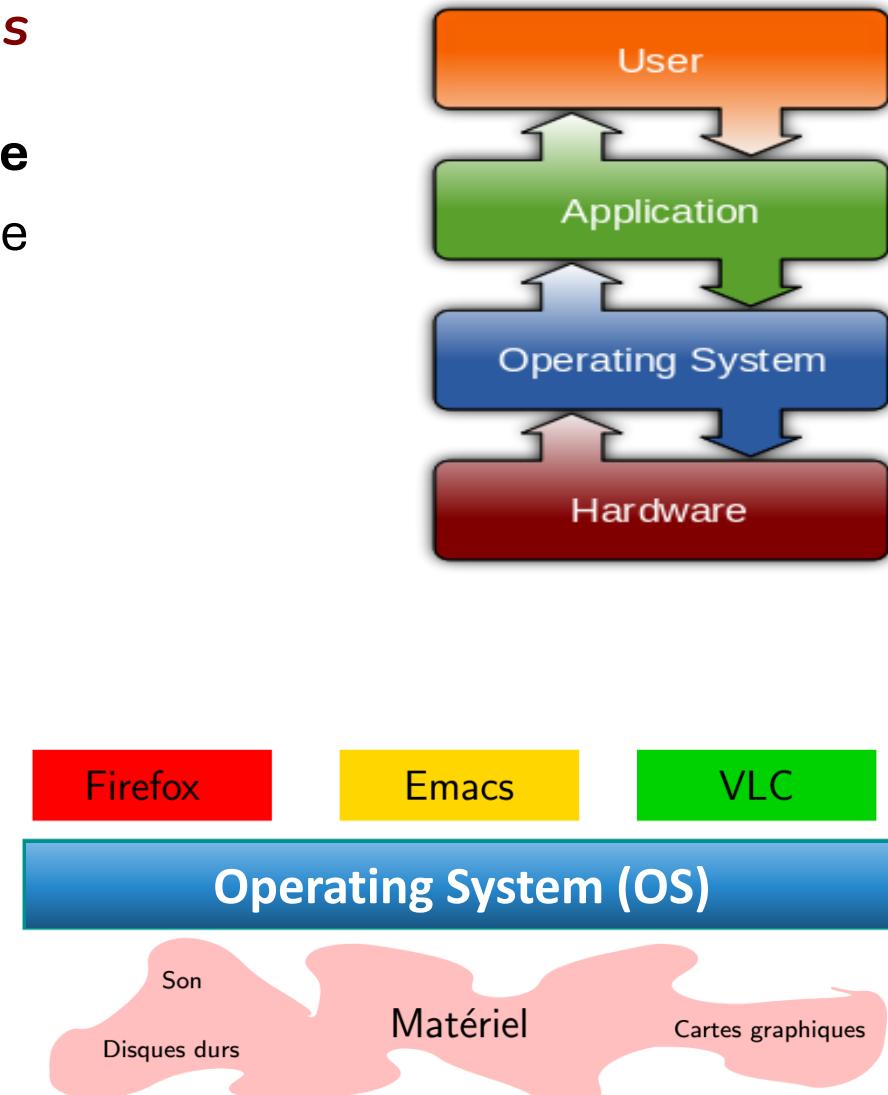
### □ The Foundation of Modern Computing: *Operating Systems*

The **operating system** acts as a **visual and graphical interface** between these electronic components of the computer and the user (and the applications used).

#### What is an Operating System ?

**Software between applications and hardware**

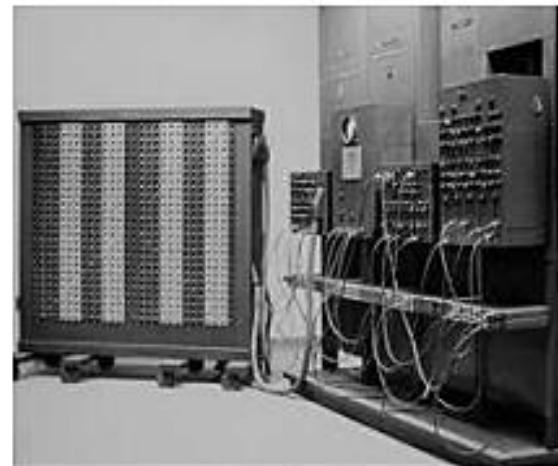
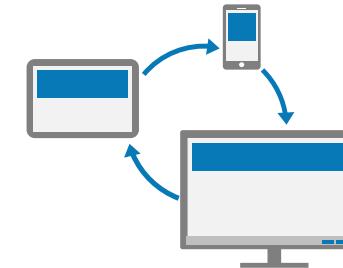
- Provides a **unified interface** for applications & simplified user experience
- Correctly and optimally manages (and protects) computer resources (Memory, CPU, Peripherals)
- Introduction of abstractions such as **Processes, Files, Virtual Memory**, etc.



## ❖ OPERATING SYSTEMS

### □ The Foundation of Modern Computing: *Operating Systems*

“They’re Everywhere !”...



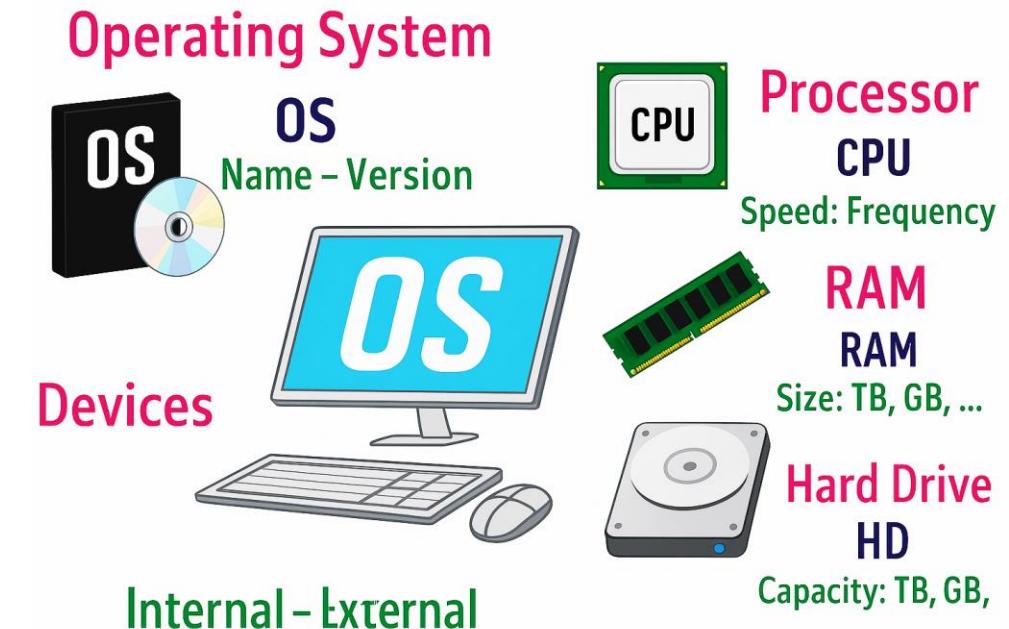
*The ENIAC Today*



## ❖ OPERATING SYSTEMS

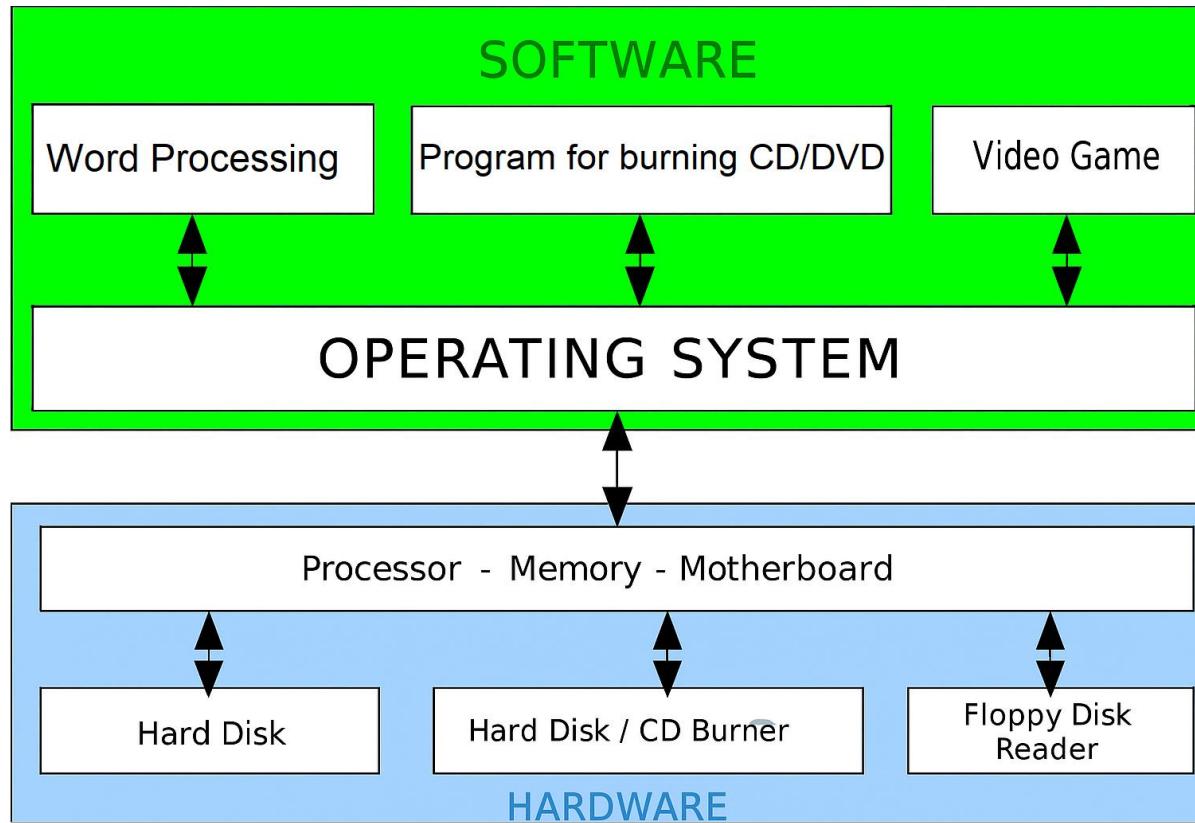
### □ Fundamental role of an Operating System

- The **essence of a computer**: a **motherboard** with a **processor** and **memory**.
- The **operating system** is a crucial program that facilitates interaction with the processor, triggering operations and calculations decided by the user.
- The **language** used to write operating system code **must be compatible with the type of processor** present in the **computer**, ensuring smooth execution of user instructions.



## ❖ OPERATING SYSTEMS

### ❑ Fundamental role of an Operating System



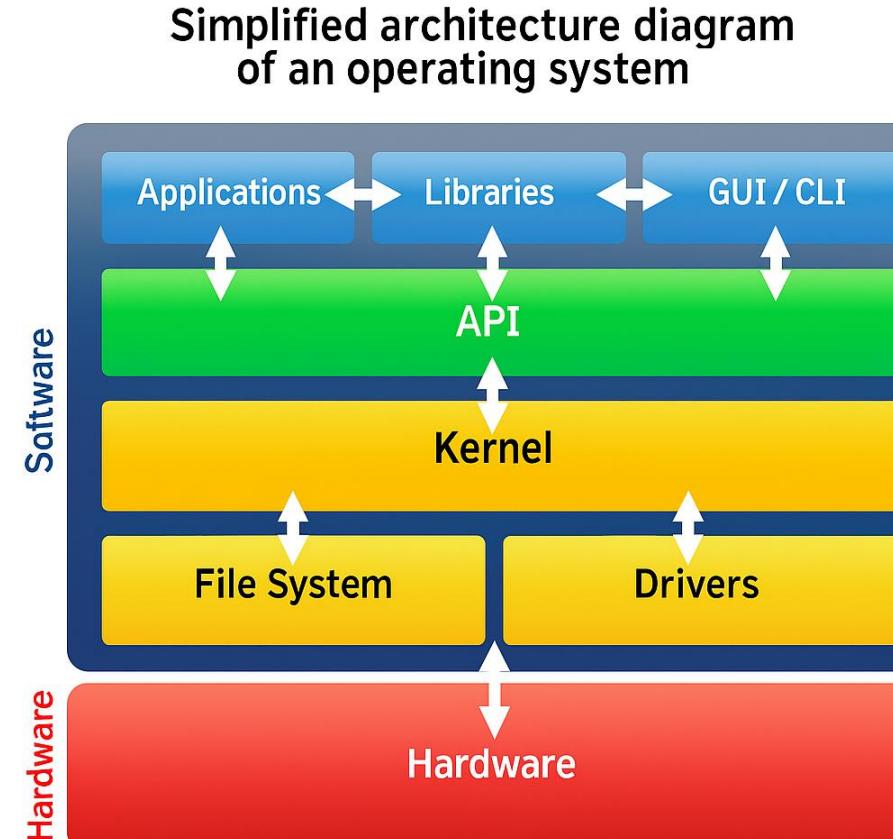
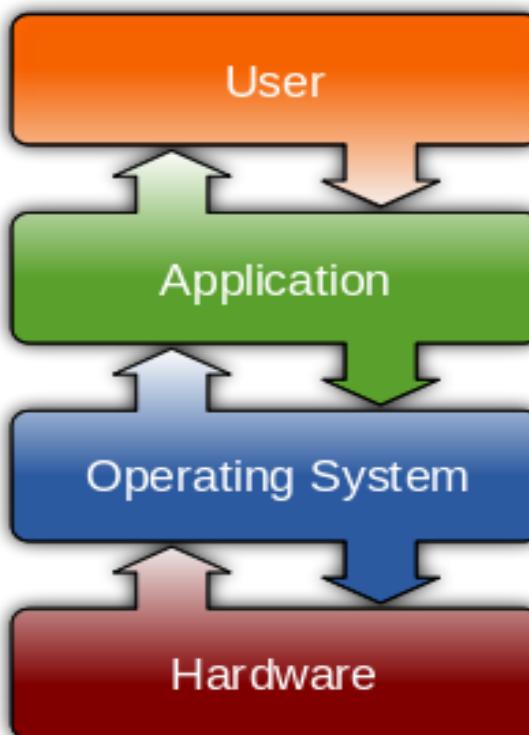
**Software**

**Hardware**



## ❖ OPERATING SYSTEMS

### □ Architecture of an Operating System



An **API — Application Programming Interface** — is a set of **rules, functions, and protocols** that allows software components to **communicate** with each other.

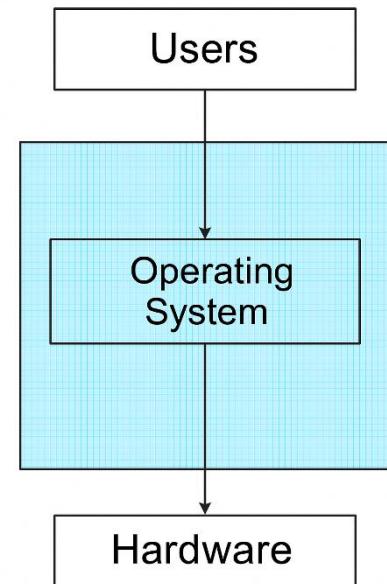
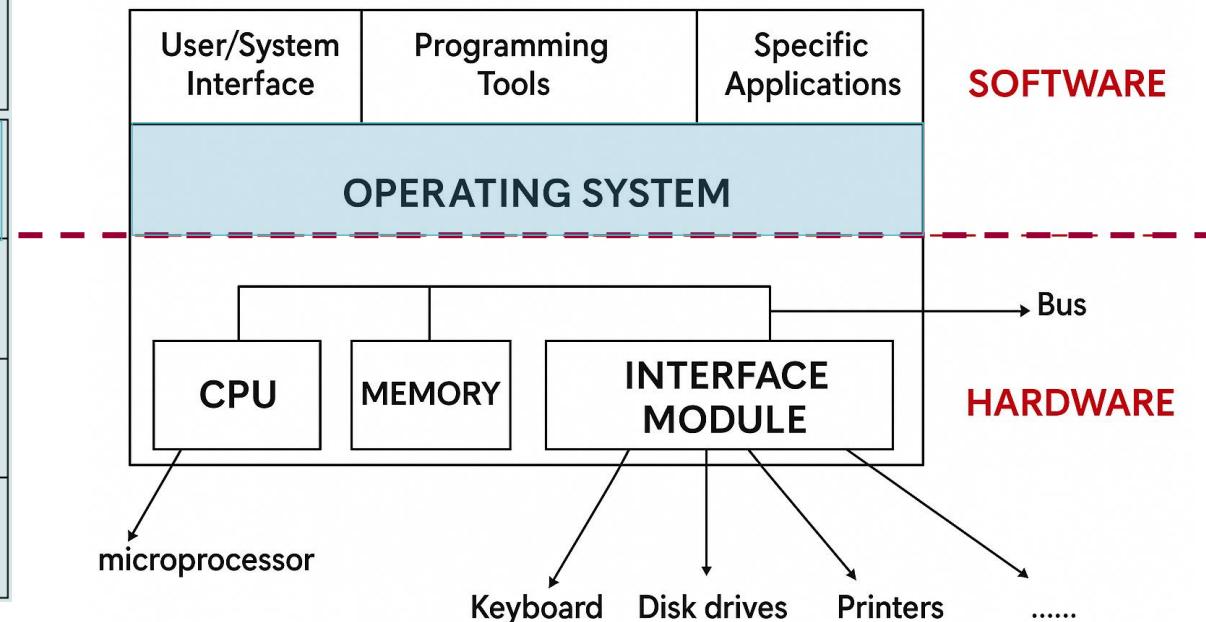
In the context of an **OS** :

- The **API** acts as a **bridge** between **applications** (*user programs*) and the **operating system's Kernel**.
- Instead of directly accessing hardware or kernel-level services, applications use **API calls** to request services like **File management, Memory allocation, or Process control**.

## ❖ OPERATING SYSTEMS

### □ Architecture of an Operating System

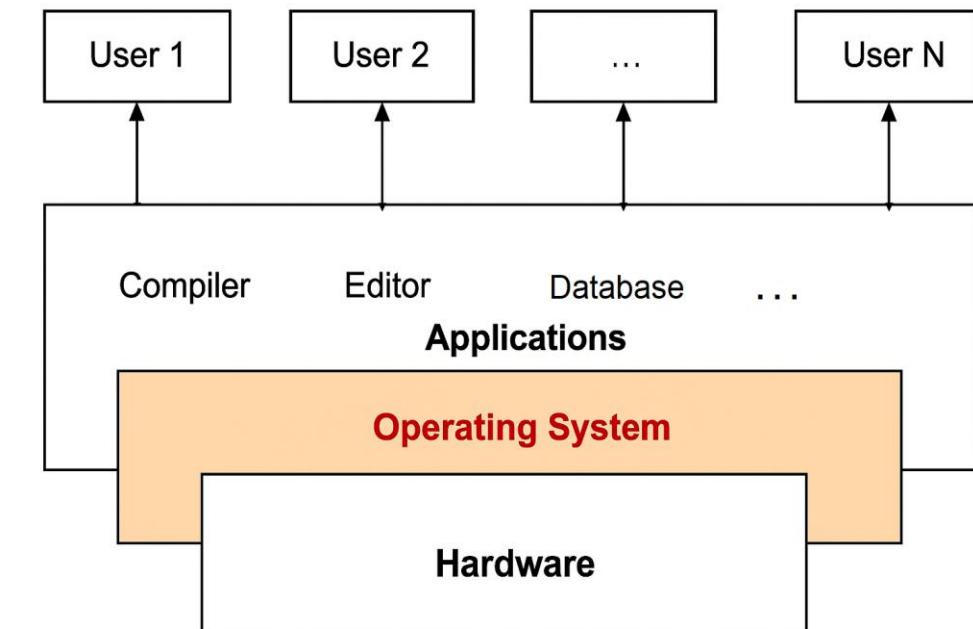
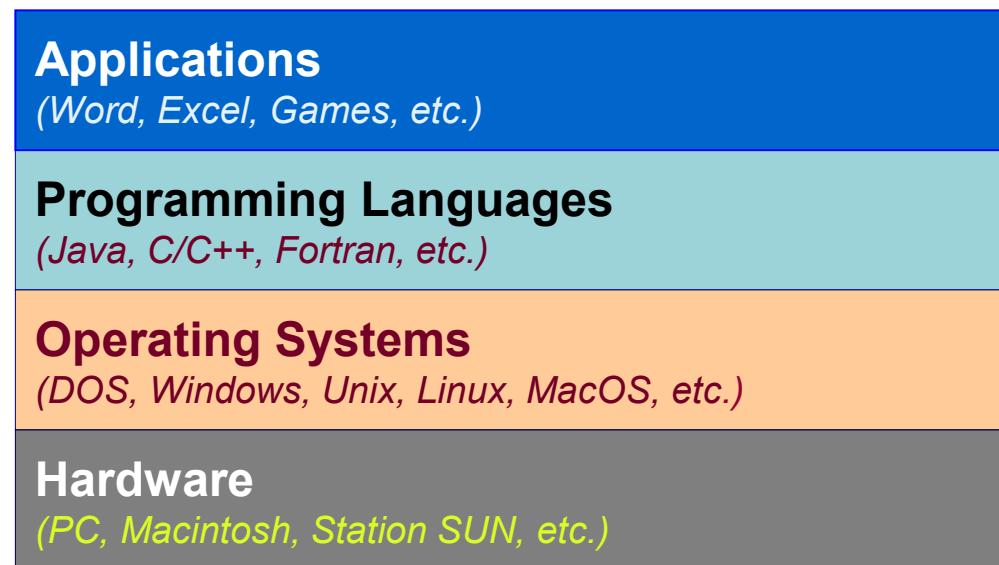
Banking system	Airline reservation	Web browser
Compilers	Editors	Command interpreter
Operating system		
Machine language		
Microprogramming		
Physical devices		



**Hardware, System programs, Application programs**

## ❖ OPERATING SYSTEMS

### □ Architecture of an Operating System



## ❖ OPERATING SYSTEMS

### □ Functional layers

**Micro-programming** is a low-level approach used in computer architecture to implement the **instruction set of a CPU**.

It involves using a sequence of micro-instructions to control the internal operations of the processor.

It allows for the **implementation of complex machine instructions** by breaking them down into smaller, manageable micro-operations.

#### User Programs

- Application Programs  
editors, spreadsheets, browsers, etc.
- System Programs  
assemblers, compilers, linkers/loaders, etc.

#### Operating System (Kernel, ... )

- Manages resources and hardware
- Provides core functions such as:  
process, memory, file management

#### Machine Instructions

The low-level commands executed directly by the CPU

- Expressed in binary or assembly language

#### Microprogramming

- A lower control layer that defines how machine instructions are executed
- Implements the CPU's control logic through microinstructions

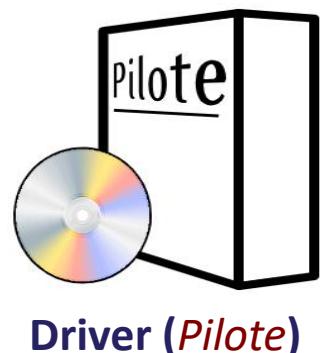
#### Hardware

## ❖ OPERATING SYSTEMS

### ❑ Interaction between the operating system and computer components

- When an operating system is installed on a computer, its first task is to **identify the components and peripherals connected to the motherboard.**
- To understand how to effectively use these elements, such as the processor, RAM and hard disk, the operating system needs a kind of “**User Manual**”.
- This guide comes in the form of **small programs** called **drivers**, which it installs or asks to install when each device is detected.
- With these drivers, the operating system **recognizes each computer component for optimal use.**

The **Bridge** between  
**Hardware** and the  
**Operating System**



 OPERATING SYSTEMS Operating System and Programs

- Once the operating system has correctly configured all of the computer's components and peripherals, it allows the user to install other programs tailored to their specific needs.
- The operating system then manages the **allocation of the computer's physical resources**, such as processor and memory time, between the various programs that are running.
- Sometimes the operating system itself can **install additional software**, but although they are installed at the same time, **they are not an integral part of the operating system**. They are considered to be software that comes with the operating system and provides a more complete user experience.

## ❖ OPERATING SYSTEMS

### □ The Graphical User Interface (GUI)

The **system's graphical user interface (GUI)** facilitates **interaction** between the user and the computer through peripherals such as the keyboard, mouse and screen.

This **user-friendly interface** makes it easy for users to perform a variety of tasks, such as installing and uninstalling programs, customizing computer settings, checking the status of peripheral connections, establishing network connections, printing documents, and much more.

In short, the **GUI** provides an **intuitive, user-friendly way to interact with the computer.**

### Operating System : Programs and Data



### Components of a Graphical User Interface



## ❖ OPERATING SYSTEMS

### □ Basic functions

#### ■ Process Management

- Execution Sequence (**Process**)
- Hardware Event Management

#### ■ Information Management

- Memory Management
- File Management

#### ■ Communication management

- Input/Output Management
- Network Management

#### ■ Security and Protection

- Information Security and Protection
- Resource Security and Protection

Physical Component	Logical Entity
Processor	Process
Main Memory	Virtual Memory
Disk	File
Screen, Keyboard, Mouse, Printer, Network	Window

## ❖ OPERATING SYSTEMS

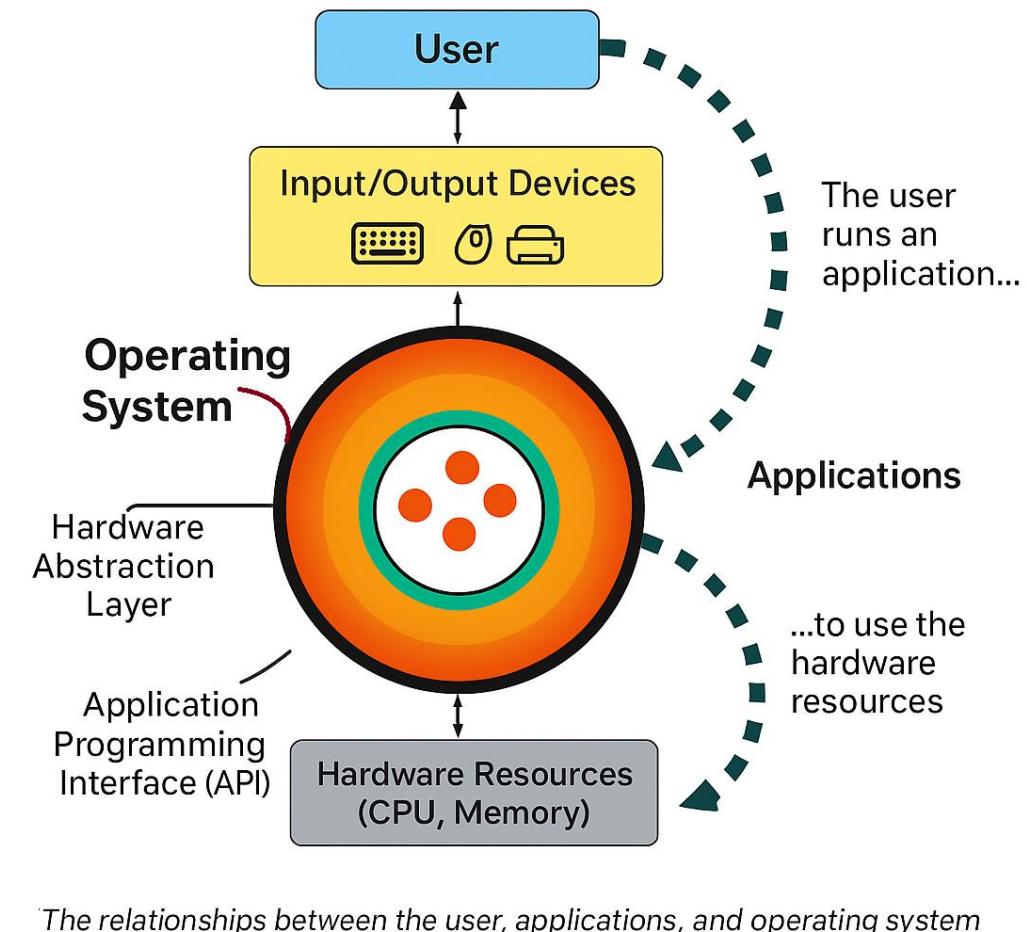
### □ Different types

- **Single task system**
  - Manage only one task at a time (**processor allocation for one program**).
- **Multi-tasking system**
  - Running more than one program at the same time on the same machine.
  - **Multi-programming**
- **Single user system**
  - Manages a single user.
- **Multi-user system**
  - Manages multiple users sharing the same hardware resources (e.g. web server, etc.).
- **Multi-processor system**
  - Use of multiple CPUs
  - Manages parallel activities
- **Distributed system** : Multiple autonomous computers connected in a network work together to perform well-defined activities.
- **Grid system (GRID)** : A large number of interconnected resources.

## ❖ OPERATING SYSTEMS

### □ System Calls

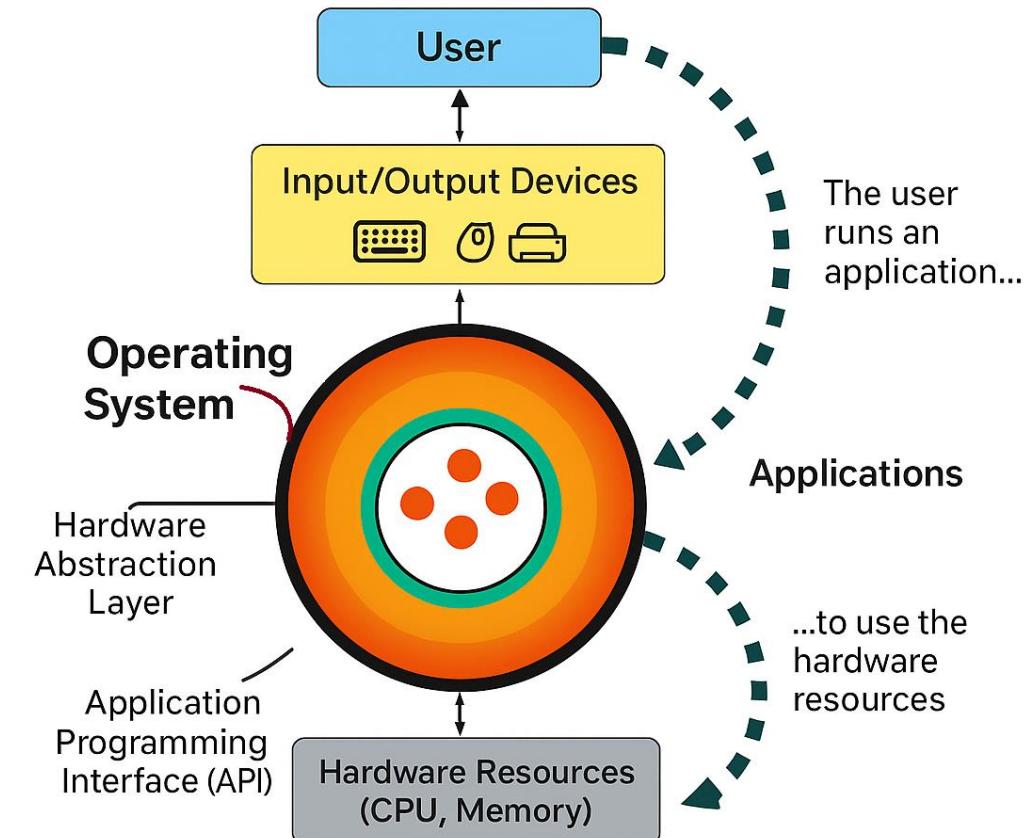
- A system call **is a function provided by the kernel and used by programs running in user space (in other words, all programs distinct from the kernel).**
- The **kernel's role** is to manage hardware resources and provide programs with **a uniform interface** for accessing these resources.
- **Some classic system calls :**
  - ✓ *open; read; write; close* to manipulate file systems,
  - ✓ *alloc; free* to allocate and deallocate memory.



## ❖ OPERATING SYSTEMS

### □ System Calls

- On most operating systems, system calls can be used as **simple functions written in C**.
- On most kernels (notably **monolithic kernels** such as the **Linux kernel**), **system calls** are implemented by a machine instruction (**interrupt**, **supervisor call**, ...) which switches the processor in the kernel to **supervisor mode** (having appropriately passed the system call parameters, for example in the registers).



*The relationships between the user, applications, and operating system*

 OPERATING SYSTEMS System Calls

- Seen from the application program, a system call is **ATOMIC**

(it may or may not have been executed in error).

Categories

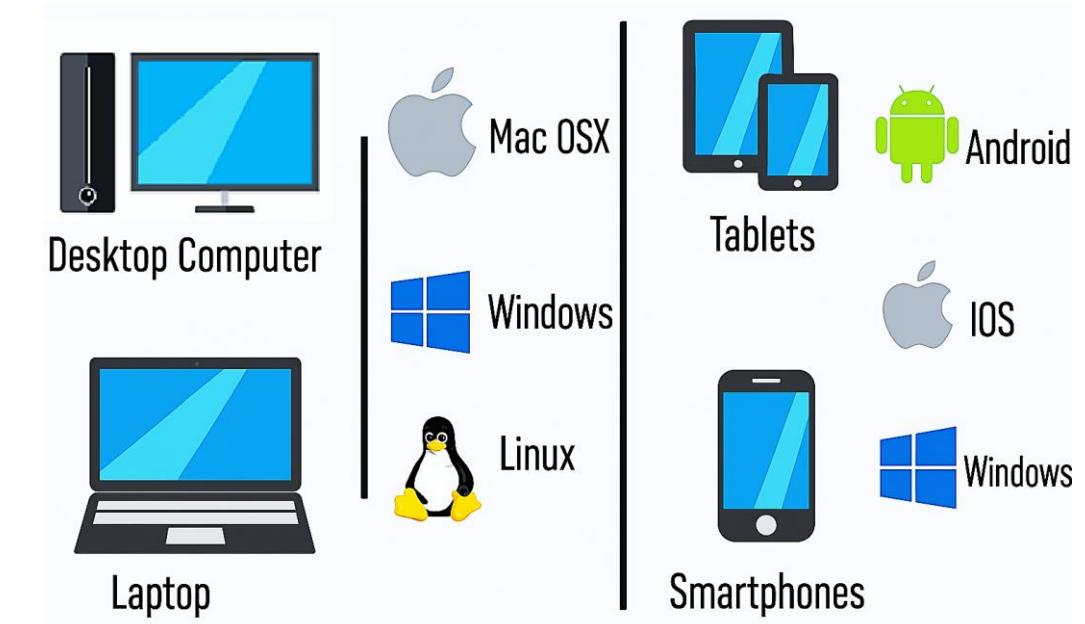
- 
1. **Process control** : load, execute, create, terminate processes, obtain, report events, free memory, etc.
  2. **File handling** : create, delete, open, close, read, write, reposition, etc.
  3. **Device management** : request, release, obtain, attach, etc.
  4. **Information maintenance** : get, set time or date, set system data
  5. **Communications** : create, delete communication connections, send, receive messages, transfer status information, etc.

## ❖ OPERATING SYSTEMS

### ❑ Most popular Operating Systems



Processors **Intel** – Processors **AMD** – Processors **IBM**



Operating systems (**MacOS, Windows, Linux, Android**) and different types of computer

## ❖ OPERATING SYSTEMS

### ❑ Most popular Operating Systems



#### Microsoft Windows

1.0 - 3.x - 95 - 98 - Me - NT - 2000 - XP - 2003 - Vista - Win 7 - Win 8 – Win 10 – **Win 11**



#### GNU/Linux

Debian - Fedora - Gentoo - Mandriva - Red Hat - Slackware - SuSE - **Ubuntu**



Mac OS

#### Apple macOS.

**Classic Mac OS:** System 5, 6, 7, 8, 9

**Mac OS X Series:** Mac OS X 10.0 – *Cheetah* ... **Mac OS X Server** – Server edition; **Kodiak** (Mac OS X 10.0)

**macOS Series:** macOS 10.7 – *Lion* ... **macOS 11 – Big Sur**



#### BSD Family (Berkeley Software Distribution)

FreeBSD – NetBSD – OpenBSD – DragonFly BSD – PC-BSD (now TrueOS)

#### Others :

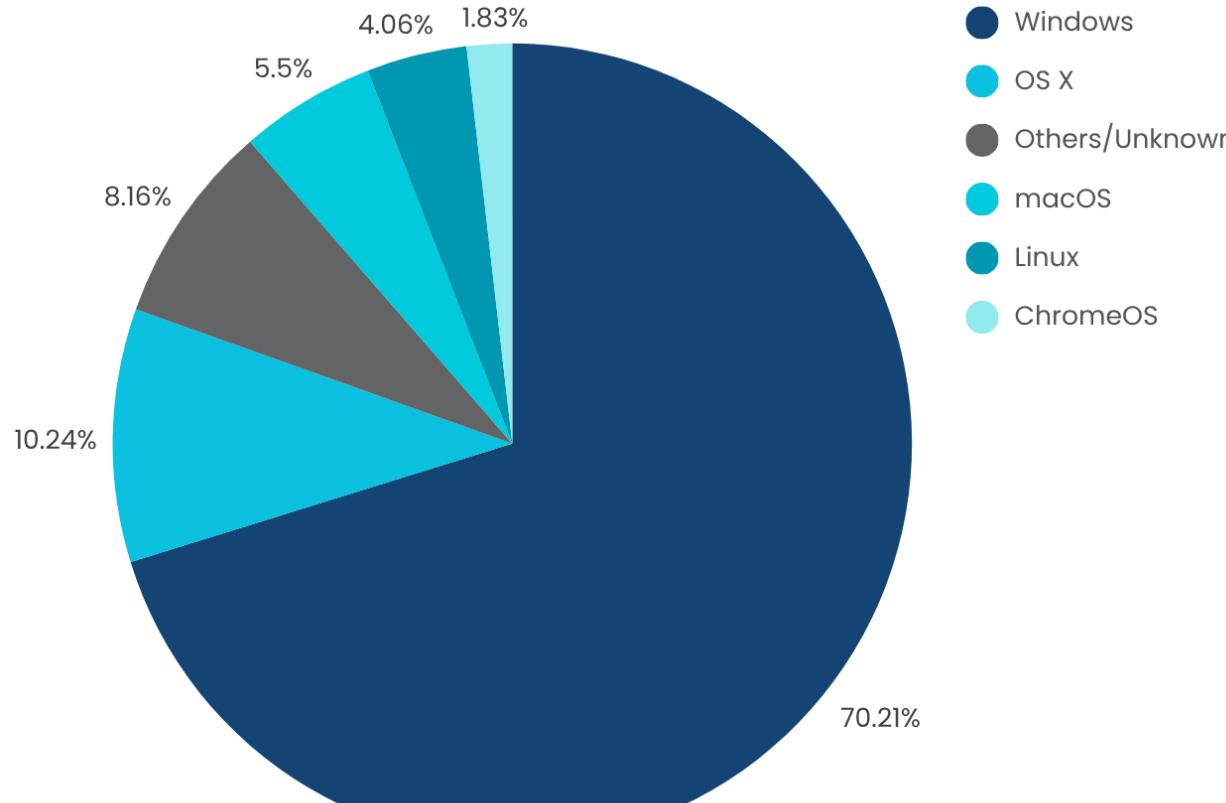
AmigaOS - BeOS - Inferno - LynxOS - Haiku - OS/2 - QNX - Solaris - UNIX

MVS - OS/360 - OS/390 - OS/400 - Plan 9 - ReactOS - VMS - ZETA - FreeDOS

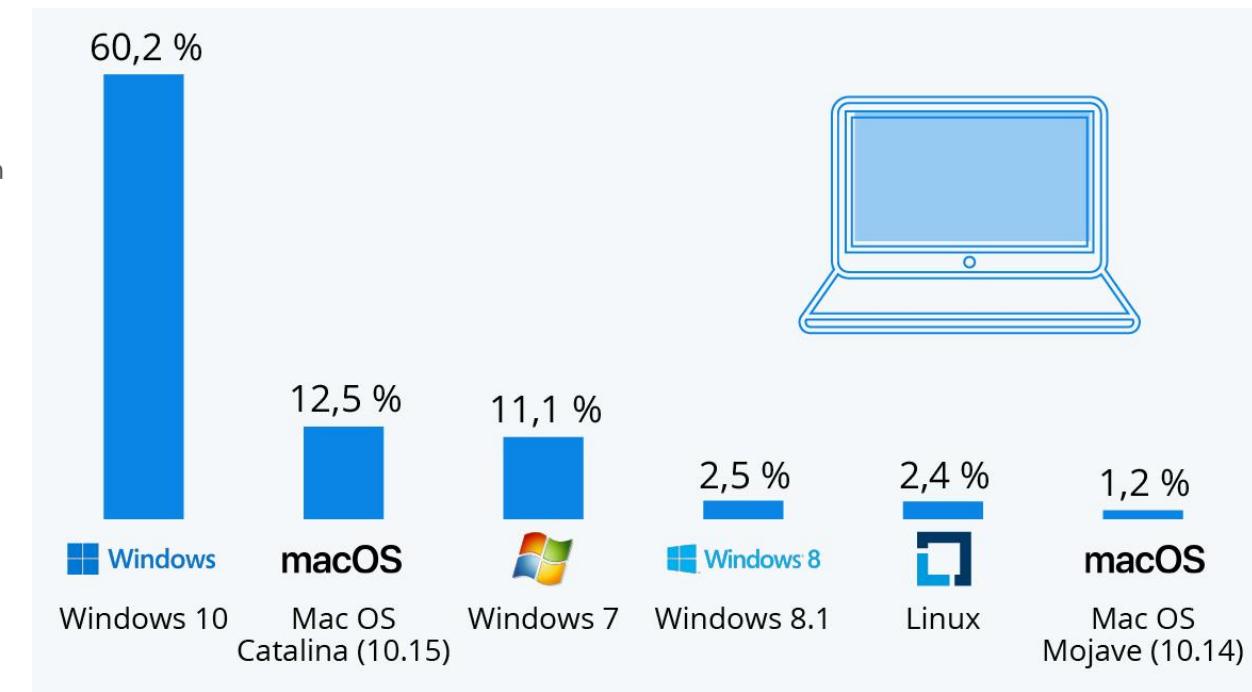
## ❖ OPERATING SYSTEMS

### □ Most popular Operating Systems

*Global Desktop OS usage as of May 2025*



*Global market share of PC OS – September 2021*



\* basée sur les pages vues.

Source : StatCounter



<https://www.statista.com/>

**statista**



**THANK YOU for your attention!**



**Questions ?**



المدرسة الوطنية العليا في الأمان السيبراني  
NATIONAL SCHOOL OF CYBERSECURITY



For more information about my research works, **Contact Information:**

**Dr. Sassi BENTRAD**

LISCO Laboratory : <http://lisco.univ-annaba.dz/>

☎ : +213 ...

✉ : sassi.bentrad.enscs@gmail.com // sassi.bentrad@enscs.edu.dz

LinkedIn : [www.linkedin.com/in/sassi-bentrad/](https://www.linkedin.com/in/sassi-bentrad/)

Website : <http://www.bentrad-sassi.sitew.com/>

**ORCID**

Connecting Research  
and Researchers

ID : [orcid.org/0000-0002-7458-8121](https://orcid.org/0000-0002-7458-8121)

**RESEARCHERID** : [A-9442-2013](https://publons.com/researcher/A-9442-2013)

**SCOPUS** Author ID : [44461052600](https://www.scopus.com/authid/detail.uri?authorId=44461052600)

